Efficient SMP-Aware MPI-Level Broadcast over InfiniBand's Hardware Multicast

Amith R. Mamidala, Lei Chai, Hyun-Wook Jin and Dhabaleswar K. Panda

Department of Computer Science and Engineering
The Ohio State University

{mamidala,chail, jinhy,panda}@cse.ohio-state.edu

Presentation Outline

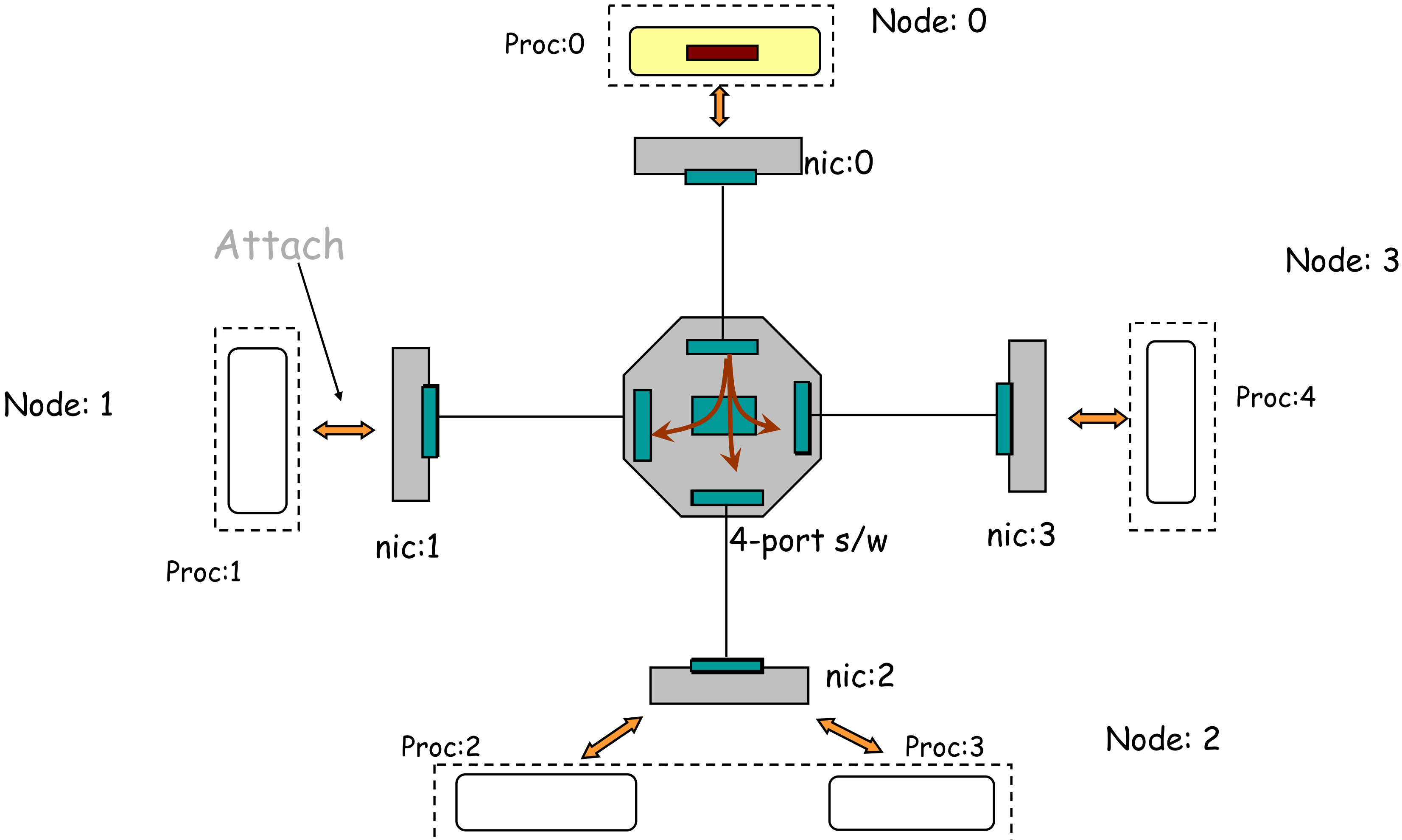
- Introduction & Background
- Motivation
- Design
- Performance Evaluation
- Conclusions & Future Work

Introduction

- Recent Advances in cluster computing
  - Size of clusters reaching tens of thousands of nodes
  - Multi-core Architecture
    - 4 to 8 cores already available
    - foresee higher process density/node (upto 16 to 32)
- InfiniBand (IBA)
  - Widely being deployed to build large-scale clusters
  - Offers many advanced features for efficient and scalable performance
    - H/W Multicast, SRQ etc.
- MVAPICH (MPI over IBA)
  - Offers many features
  - Shared Memory Channel
    - Low latency compared to network
    - Intra-node point-to-point operations
  - Collectives
    - H/W Multicast, RDMA
- MPI_Bcast
  - Important collective operation
  - Scalable , Low latency design over H/W multicast

    (J. Liu, A. Mamidala, D.K. Panda, "Fast and Scalable MPI-Level Broadcast using InfiniBand's Hardware Multicast Support", IPDPS 04)

Background: MPI_Bcast over H/W Multicast

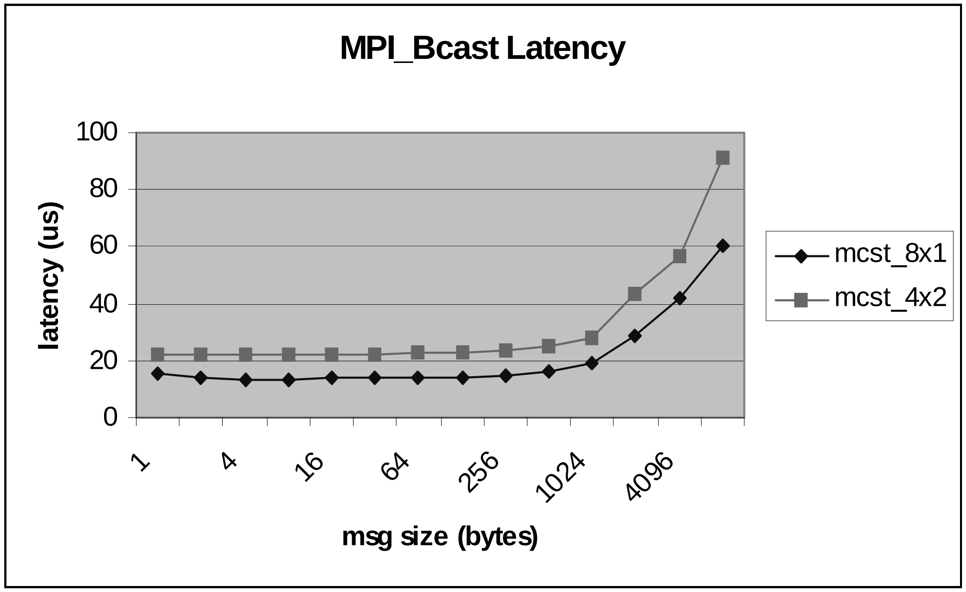


Presentation Outline

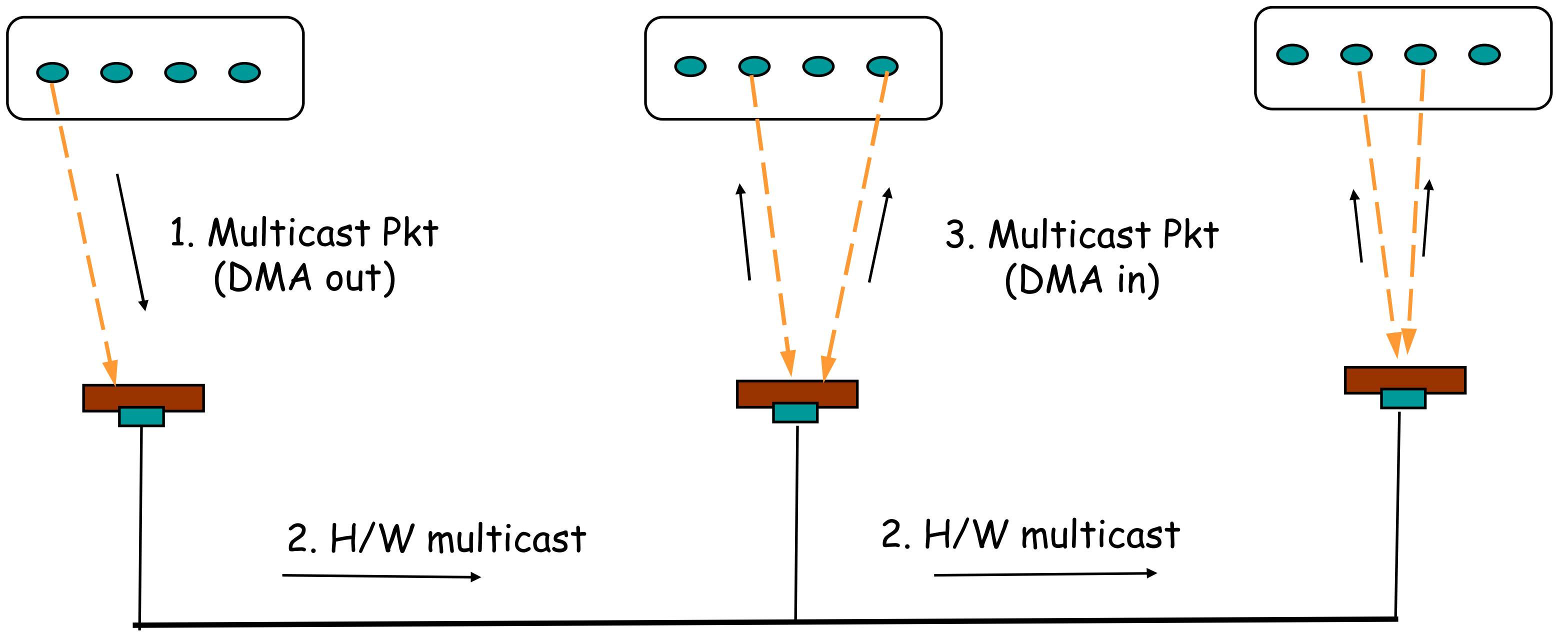
- Introduction & Background
- Motivation
- Design
- Performance Evaluation
- Conclusions & Future Work

Motivation

- Original solution not optimal for higher process count



Motivation



- Cost incurred for each multicast pkt
  - Replication at the nic
  - DMA transaction
- Significantly affects the latency if the process density increases

Motivation

- Reliability
  - H/W multicast is unreliable
- Large message handling
  - H/W multicast in MTUs
- How to employ best communication methods within a node (Shared Memory) and across the nodes (H/W Multicast) for efficient and reliable MPI_Bcast?

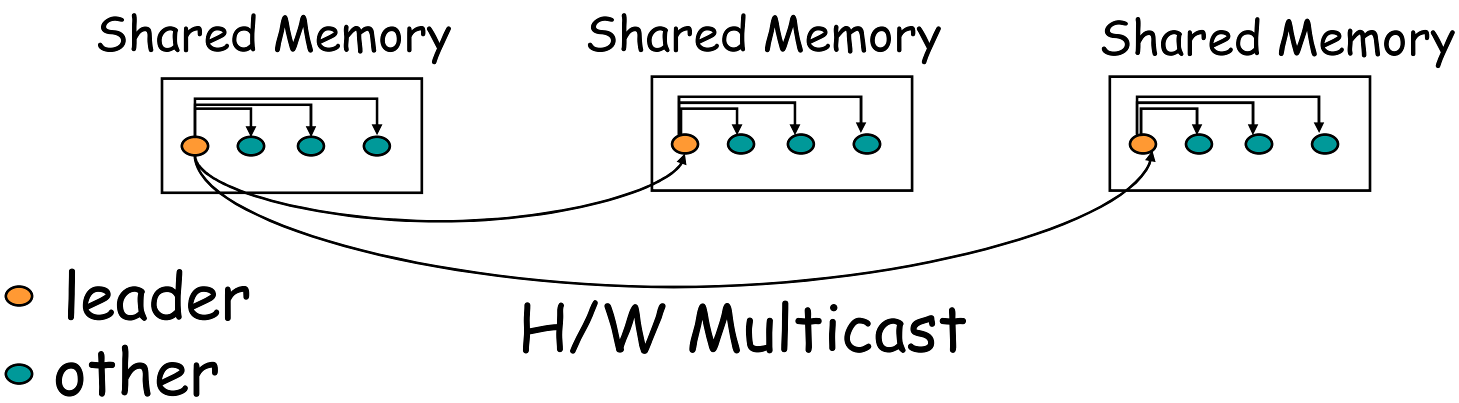
Presentation Outline

- Introduction
- Background & Motivation
- Design
- Performance Evaluation
- Conclusions & Future Work

Design

- Direct Multicast into Shared Memory
  - Complex to implement
    - Message notification
      - Completion notified to only the "attached" processes
    - Buffer Management
    - Reliability
- Leader-based Approach
  - A designated process chosen as the leader
  - Leader handles
    - H/W multicast packet delivery/reception
    - reliability
    - large message handling
  - Shared memory to distribute the multicast packet to the remaining nodes
  - Simple solution
  - Taken this approach

Leader-based design



- leader attaches to the multicast group
- responsible for handling reliability
- forwards the multicast pkts to other nodes

Choosing Leader

- Fixing the leader doesnot always perform well
  - leader arrives late
  - Other nodes depend on leader for packet forwarding
- Dynamic Attach Policy
  - Choosing leader based on certain criterion
  - Chosen leader dynamically attaches to H/W multicast group

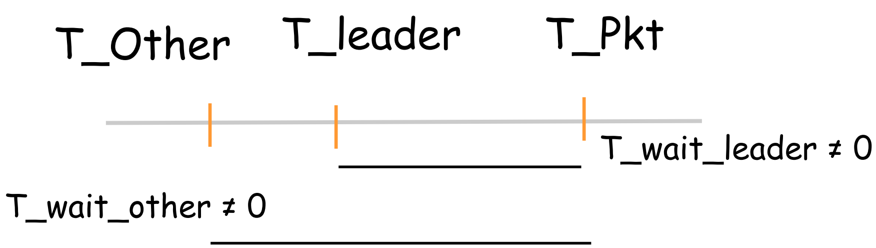
Dynamic Attach Policy

- Basic idea: Using Average Wait Time
  - Non-leader process selectively attaches/detaches based on this time
  - Average Wait Time relative to the leader
  - Average Wait Time = (Total Wait Time)/ (# Broadcast operations so far)
- Computing Total Wait Time
  - Depends upon the order of arrival of
    - Multicast packet
    - Leader process
    - Non-Leader process

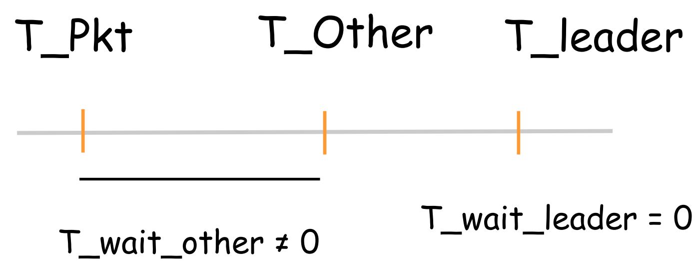
Total Wait Time

Relevant Cases:

Case 1:



Case 2:



Use a global flag visible to all processes to eliminate case 1

Presentation Outline

- Introduction
- Background & Motivation
- Design
- Performance Evaluation
- Conclusions & Future Work

Performance Evaluation

- MPI_Bcast Latency test:
  - Maximum of the latency for each of non-root nodes
- Two cases considered:
  - All processes are sychronized
  - Leader process arrives late
- Three schemes for comparison
  - mcst_smp: new design with SMP-Aware multicast
  - mcst_nosmp: old design with non SMP-Aware multicast
  - original_ptp: original pt-to-pt design
- Incorporated into MVAPICH (OSU's MPI over IBA)

OSU MPI over InfiniBand

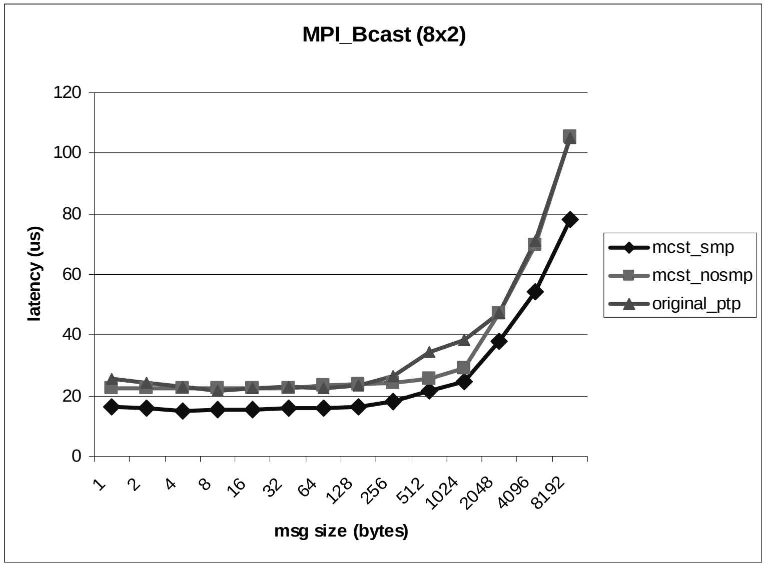
- High Performance Implementations
  - MPI-1 (MVAPICH)
  - MPI-2 (MVAPICH2)
- Open Source (BSD licensing)
- Has enabled a large number of production IB clusters all over the world to take advantage of IB
  - Largest being Sandia Thunderbird Cluster (4000 node with 8000 processors)
- Have been directly downloaded and used by more than 340 organizations worldwide (in 33 countries)
  - Time tested and stable code base with novel features
- Available in software stack distributions of many vendors
- Available in the OpenIB/gen2 stack
- More details at

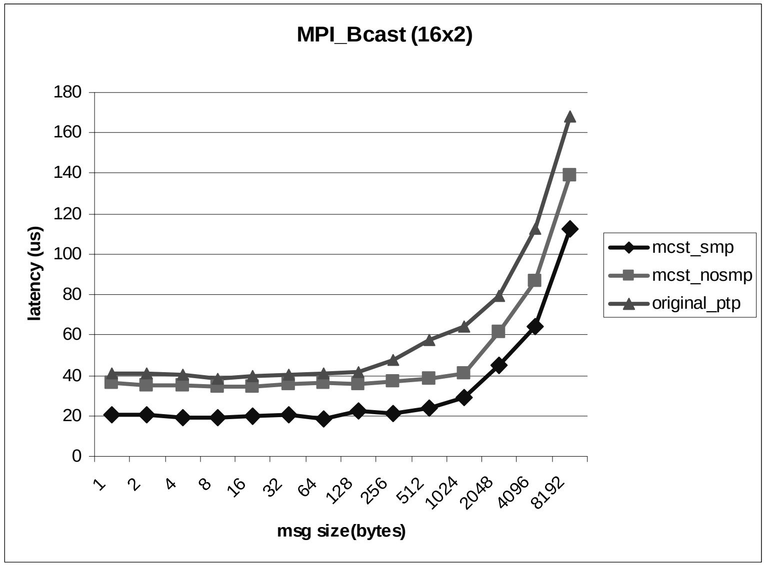
  http://nowlab.cse.ohio-state.edu/projects/mpi-iba/

Evaluation Testbed

- Cluster A:
  - 16 Intel Xeon 3.0 GHz processors
  - PCI-X 64 bit, 133 MHz bus
  - MT23108 Mellanox HCAs
- Cluster B:
  - 8 dual Intel Xeon EM64t 3.2 GHz processors
  - PCI-Express Interface
  - MT25128 Mellanox HCAs
- Cluster C:
  - 2 Quad Opterons
  - PCI-X interface
  - MT23108 Mellanox HCAs
- InfiniScale 24 port switch
- OpenSM: Subnet Manager

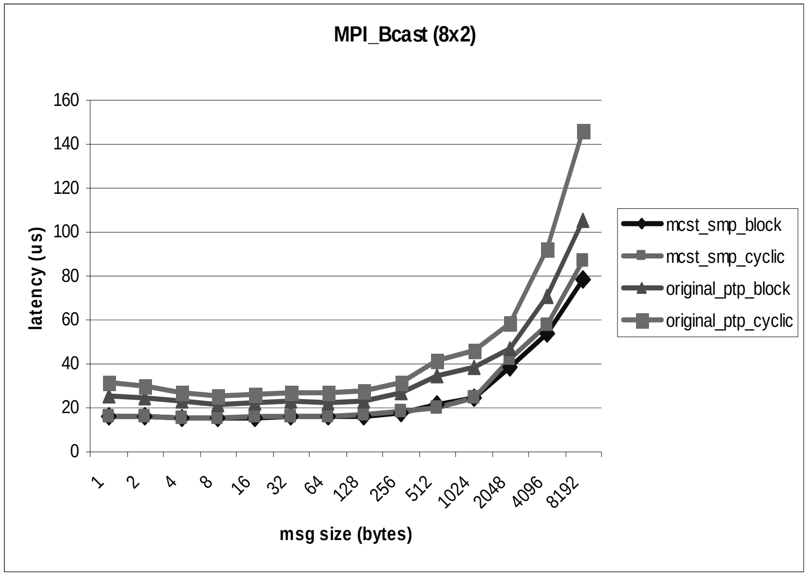
MPI_Bcast Latency

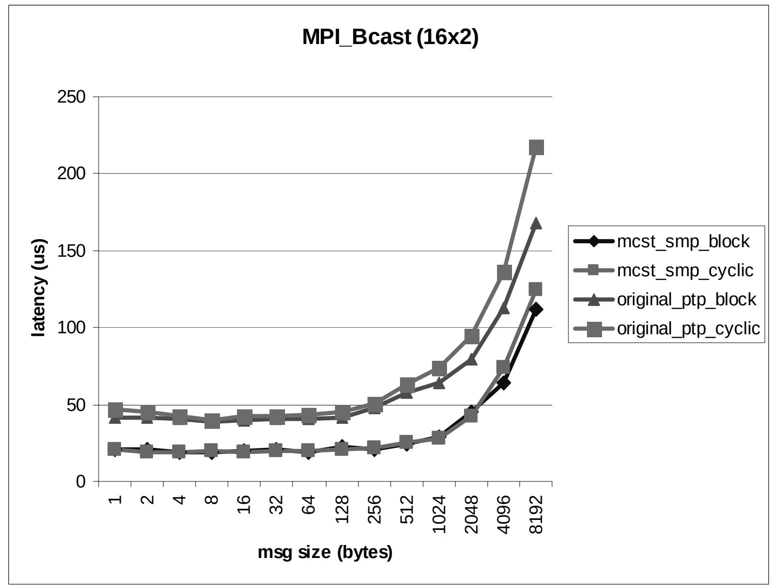




-Block distribution to scatter processes
-Improves latency by a factor of 2.18 and 1.8 compared to original_mcst and smp_nomcst

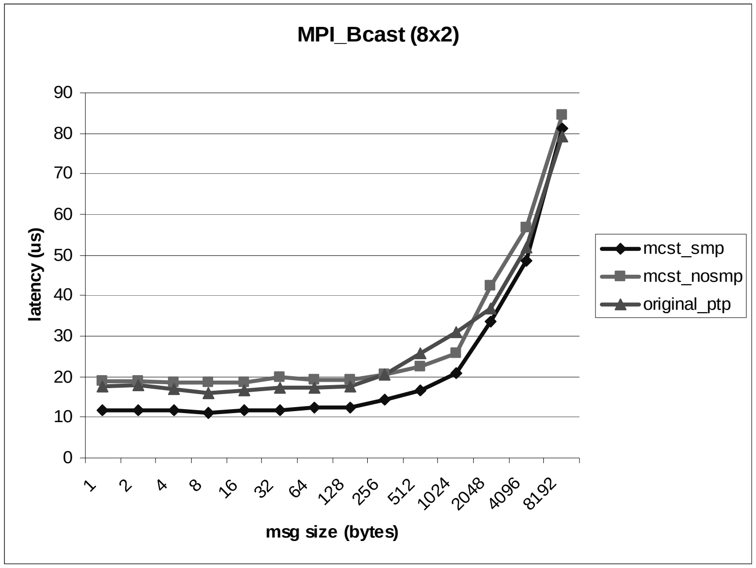
Different configurations

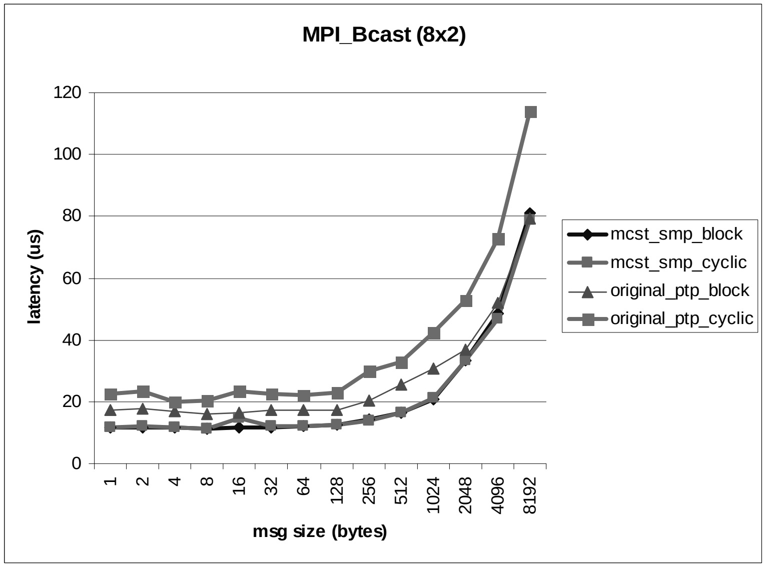




- Cyclic or Block distributions have no affect for smp_mcst design
-Impact original_mcst, Intra-node messages delivered first

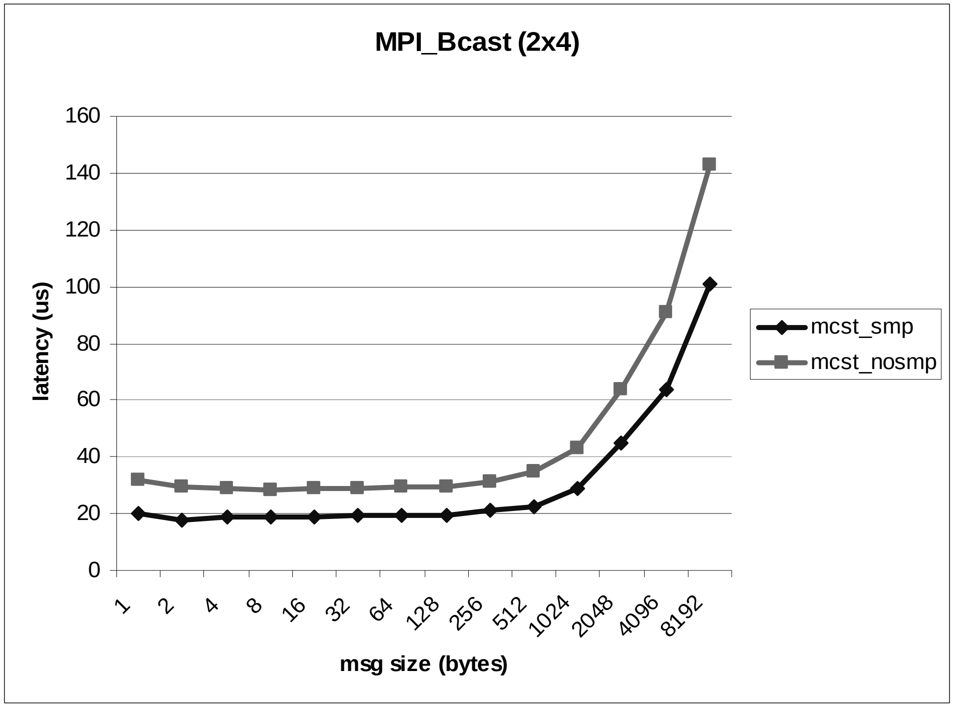
MPI_Bcast Latency





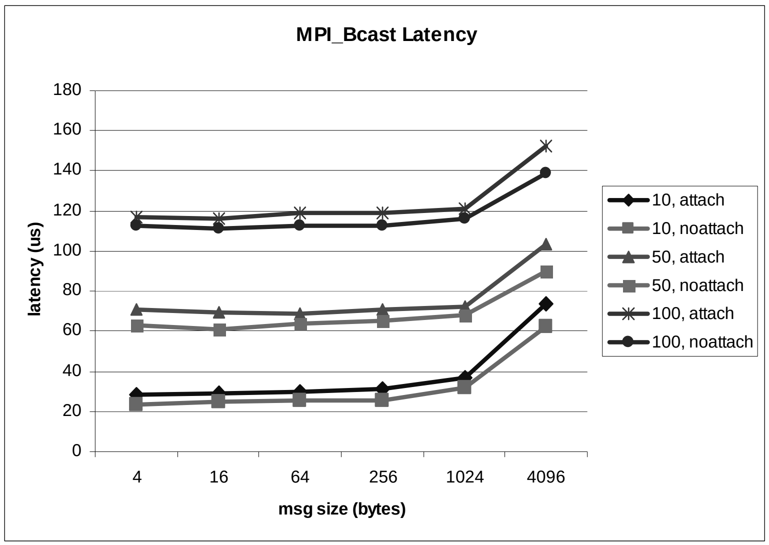
- smp_mcst improves performance by a factor upto two
- process distribution: no impact for smp_mcst peformance
- block does better than cyclic for original_mcst

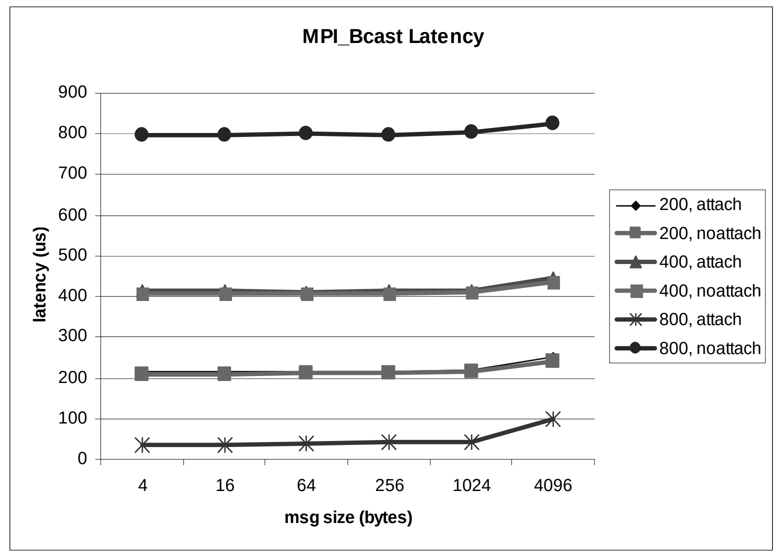
MPI_Bcast Latency



- Performance improvement upto 1.7 for Quad Opterons

Impact of Dynamic Attach Policy





- Threshold of 500 us (no. of processes * attach_latency: 2x250)

Presentation Outline

- Introduction
- Background & Motivation
- Design
- Performance Evaluation
- Conclusions & Future Work

Conclusions & Future Work

- Efficient SMP-Aware MPI_Bcast using IBA's H/W multicast support
- Leader-based design,
- Dynamic Attach Policy proposed to mitigate skew effects
- Evaluated performance with different configurations
- Future work: Evaluation with higher SMP-way systems
- Integrated into MVAPICH

Acknowledgements

Our research is supported by the following organizations

- Current Funding support by

- Current Equipment support by

Web Pointers

Network Based Computing Laboratory

http://www.cse.ohio-state.edu/~panda/

http://nowlab.cse.ohio-state.edu/

MVAPICH Web Page

http://nowlab.cse.ohio-state.edu/projects/mpi-iba/

Questions ?